AMENDMENTS TO THE CLAIMS

Please amend Claims 1 and 15 as follows, without prejudice or disclaimer to continued examination on the merits:

1. (Currently Amended): A computer implemented mechanism for dynamically constructing a network element management system, comprising:

executing a core application, the core application comprising NE element-independent Version Independent ~~(VINI)~~ functionality that can be invoked to interact with any network element, wherein NE element-independent Version Independent functionality comprises the core application containing no code specific with any network element;

receiving an indication of a particular network element to be managed;

obtaining a description of the particular network element, which specifies one or more characteristics of the particular network element;

accessing, based upon the description, a set of one or more element-dependent modules, the set of element-dependent modules comprising functionality for invoking one or more management services provided by the particular network element; ~~and~~

dynamically incorporating at least a portion of the set of element-dependent modules with the core application to derive a management system capable of managing the particular network element; and

managing the particular network element with the management system with the set of element-dependent modules.

2. (Original): The method of claim 1, wherein the set of element-dependent modules comprises a first proxy corresponding to a first management service provided by the particular network element, the first proxy comprising functionality for converting a generic service invocation from the core application into a specific invocation of the first management service.

3. (Original): The method of claim 2, wherein the first proxy implements a common interface implemented by all proxies, and wherein the core application interacts with the first proxy via the common interface.

4. (Previously Presented): The method of claim 3, wherein dynamically incorporating the set of element-dependent modules comprises:

instantiating the first proxy to give rise to a first proxy instance; and

incorporating the first proxy instance with the core application.

5. (Original): The method of claim 3, wherein the first proxy manages one or more managed object classes, and wherein the set of element-dependent modules further comprises the one or more managed object classes.

6. (Original): The method of claim 5, wherein dynamically incorporating the set of element-dependent modules comprises:

loading the one or more managed object classes.

7. (Original): The method of claim 5, wherein each managed object class implements a second common interface implemented by all managed object classes.

8. (Original): The method of claim 7, where each managed object class comprises functionality for converting a raw data type used by the first management service into a generic data type used by the core application.

9. (Original): The method of claim 7, where each managed object class comprises functionality for converting a generic data type used by the core application into a raw data type used by the first management service.

10. (Original): The method of claim 1, wherein the set of element-dependent modules comprises a service descriptor which provides a description of at least one particular management service provided by the particular network element, the service

descriptor specifying one or more managed object classes managed by the particular management service.

11. (Original): The method of claim 10, wherein the set of element-dependent modules comprises a user interface screen associated with a particular managed object class, and wherein the user interface screen comprises fields for rendering data corresponding to attributes of the particular managed object class.

12. (Original): The method of claim 11, wherein the set of element-dependent modules further comprises a screen descriptor associated with the user interface screen, the screen descriptor specifying a mapping between the fields of the user interface screen and the attributes of the particular managed object class.

13. (Original): The method of claim 1, wherein the characteristics of the particular network element comprise an indication of an element type, and an indication of an element version.

14. (Original): The method of claim 1, further comprising:

receiving an indication of a second network element to be managed;

obtaining a second description of the second network element, which specifies one or more characteristics of the second network element;

accessing, based upon the second description, a second set of one or more element-dependent modules, the second set of element-dependent modules comprising functionality for invoking one or more management services provided by the second network element; and

dynamically incorporating at least a portion of the second set of element-dependent modules with the core application to derive a management mechanism capable of managing both the particular network element and the second network element.

15. (Currently Amended): A computer readable medium, comprising:

instructions for causing one or more processors to give rise to a core application, the core application comprising element-independent functionality that can be invoked to interact with any network element; and wherein the core application comprises no code specific with any network element;

instructions for causing one or more processors to receive an indication of a particular network element to be managed;

instructions for causing one or more processors to obtain a description of the particular network element, which specifies one or more characteristics of the particular network element;

instructions for causing one or more processors to access, based upon the description, a set of one or more element-dependent modules, the set of element-dependent modules comprising functionality for invoking one or more management services provided by the particular network element; ~~and~~

instructions for causing one or more processors to dynamically incorporate at least a portion of the set of element-dependent modules with the core application to derive a management mechanism capable of managing the particular network element; and

instructions for causing one or more processors to manage the particular network element with the set of element-dependent modules.

16. (Original): The computer readable medium of claim 15, wherein the set of element-dependent modules comprises a first proxy corresponding to a first management service provided by the particular network element, the first proxy comprising functionality for converting a generic service invocation from the core application into a specific invocation of the first management service.

17. (Original): The computer readable medium of claim 16, wherein the first proxy implements a common interface implemented by all proxies, and wherein the core application interacts with the first proxy via the common interface.

18. (Original): The computer readable medium of claim 17, wherein the instructions for causing one or more processors to dynamically incorporate the set of element-dependent modules comprises:

instructions for causing one or more processors to instantiate the first proxy to give rise to a first proxy instance; and

instructions for causing one or more processors to incorporate the first proxy instance
with the core application.

19. (Original): The computer readable medium of claim 17, wherein the first proxy manages one or more managed object classes, and wherein the set of element-dependent modules further comprises the one or more managed object classes.

20. (Original): The computer readable medium of claim 19, wherein the instructions for causing one or more processors to dynamically incorporate the set of element-dependent modules comprises:

instructions for causing one or more processors to load the one or more managed object classes.

21. (Original): The computer readable medium of claim 19, wherein each managed object class implements a second common interface implemented by all managed object classes.

22. (Original): The computer readable medium of claim 21, where each managed object class comprises functionality for converting a raw data type used by the first management service into a generic data type used by the core application.

23. (Original): The computer readable medium of claim 21, where each managed object class comprises functionality for converting a generic data type used by the core application into a raw data type used by the first management service.

24. (Original): The computer readable medium of claim 15, wherein the set of element-dependent modules comprises a service descriptor which provides a description of at least one particular management service provided by the particular network element, the service descriptor specifying one or more managed object classes managed by the particular management service.

25. (Original): The computer readable medium of claim 24, wherein the set of element-dependent modules comprises a user interface screen associated with a particular managed object class, and wherein the user interface screen comprises fields for rendering data corresponding to attributes of the particular managed object class.

26. (Original): The computer readable medium of claim 25, wherein the set of element-dependent modules further comprises a screen descriptor associated with the user interface screen, the screen descriptor specifying a mapping between the fields of the user interface screen and the attributes of the particular managed object class.

27. (Original): The computer readable medium of claim 15, wherein the characteristics of the particular network element comprise an indication of an element type, and an indication of an element version.

28. (Original): The computer readable medium of claim 15, further comprising:

instructions for causing one or more processors to receive an indication of a second network element to be managed;

instructions for causing one or more processors to obtain a second description of the second network element, which specifies one or more characteristics of the second network element;

instructions for causing one or more processors to access, based upon the second description, a second set of one or more element-dependent modules, the second set of element-dependent modules comprising functionality for invoking one or more management services provided by the second network element; and

instructions for causing one or more processors to dynamically incorporate at least a portion of the second set of element-dependent modules with the core application to derive a management mechanism capable of managing both the particular network element and the second network element.

29. (Withdrawn): A management mechanism for managing a network, comprising:

a core application; and

an element-dependent module being associated with a network element in said network, said element-dependent module being invokable by said core application.

30. (Withdrawn): The management mechanism of claim 29, wherein the element-dependent module includes:

an element descriptor;

a connection manager;

a service descriptor;

a proxie;

a managed object class;

a managed object descriptor;

a user interface screen;

a screen descriptor; and

an operation descriptor.